

Cl6z & plus/4 VIC-20 TI-99/4A sinclair  
 sinclair Cl6z & plus/4 VIC-20 TI-99/4A  
 TI-99/4A sinclair Cl6z & plus/4 VIC-20  
 Cl6z & plus/4 VIC-20 TI-99/4A sinclair  
 sinclair Cl6z & plus/4 VIC-20 TI-99/4A  
 TI-99/4A sinclair Cl6z & plus/4 VIC-20  
 VIC-20 TI-99/4A sinclair Cl6z & plus/4  
 Cl6z & plus/4 VIC-20 TI-99/4A sinclair  
 sinclair Cl6z & plus/4 VIC-20 TI-99/4A  
 TI-99/4A sinclair Cl6z & plus/4 VIC-20  
 VIC-20 TI-99/4A sinclair Cl6z & plus/4  
 Cl6z & plus/4 VIC-20 TI-99/4A sinclair



# PAPER

## SOFT

# 47

IL PRIMO SETTIMANALE  
 DI SOFTWARE SU CARTA PER  
 C16 & PLUS/4 - VIC 20 -  
 TI-99/4A - SINCLAIR

Una pubblicazione della  
 J.soli editrice  
 Con la collaborazione del  
 Gruppo Editoriale Jackson

- Mini-Mancala
- Escher
- Merge
- Merge & Renumber
- Fuori strada
- Calibro
- Missione Impossibile
- Tombola

sinclair  
 plus/4 VIC-20 TI-99/4A  
 sinclair Cl6z & plus/4 VIC-20  
 VIC-20 TI-99/4A sinclair Cl6z & plus/4

# Guida per l'input dei programmi versioni C16 plus/4 - VIC 20

Notate che i listati contengono "parole" racchiuse tra parentesi graffe {}. Tali parole rappresentano caratteri di controllo come mostrato nel sottostante riquadro. Se sono precedute da un numero, questo indica il numero di volte che quel tasto deve essere premuto. Se il simbolo è sottolineato deve essere premuto contemporaneamente a SHIFT mentre se è racchiuso da [()] deve esse-

re premuto contemporaneamente al tasto COMMODORE. Inoltre, se tra parentesi si trova un carattere alfabetico "solitario" questo dovrà essere premuto contemporaneamente al tasto CONTROL.

Con questo sistema di codifica, sarà molto più agevole copiare i listati senza faticose e dubbie interpretazioni di caratteri grafici e di controllo del cursore o dei colori.

[CLR]	SHIFT	[CYN]	CTRL	[<7>]	
[HOME]	CTRL	[PUR]	CTRL	[<8>]	
[SP] "	SHIFT	[GRN]	CTRL	[F1]	
[G1"]	CTRL	[BLU]	CTRL	[F2]	
[SH]	SHIFT	[YEL]	CTRL	[F3]	
[DFS]	CTRL	[<1>]		[F4]	
[RVS]	CTRL	[<2>]		[F5]	
[OFF]	CTRL	[<3>]		[F6]	
[BLK]	CTRL	[<4>]		[F7]	
[WHT]	CTRL	[<5>]		[F8]	
[RED]	CTRL	[<6>]			

Per ricevere gli arretrati di **PAPER** compilare il seguente tagliando:



**J.soft** compilare ed inviare in busta chiusa a:  
viale Restelli, 5 - 20124 Milano - Tel. 02/6888228-683797-6880841-2-3

Nome \_\_\_\_\_

Cognome \_\_\_\_\_

Indirizzo \_\_\_\_\_

CAP \_\_\_\_\_ Città \_\_\_\_\_ Prov \_\_\_\_\_

Desidero ricevere i seguenti numeri arretrati di

**PAPER** Anno \_\_\_\_\_

**PAPER** dedicato a \_\_\_\_\_ Anno \_\_\_\_\_

a L. 2.000 cad per un totale di L.

a mezzo

☐ assegno allegato

☐ contanti allegati

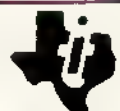


## Cl6z-plus/4

- 6** Mini-Mancala  
di R. Gaslers trad. e adatt. di F. Sarcina
- 8** Escher  
di W. Stonefall trad. e adatt. di F. Sarcina
- 11** Merge  
di H. Van Meik trad. e adatt. di F. Sarcina

## VIC-20

- 12** Merge & Renumber  
di H. Van Meik trad. e adatt. di F. Sarcina
- 14** Fuoristrada (3K Ram-Joystick)  
di S. Martini trad. e adatt. di S. Albarelli



## TI-99/4A

- 18** Calibro (BASIO)  
di C. Regena trad. e adatt. di E. Re Garbagnati

## Sinclair Spectrum

- 25** Missione Impossibile (16/48K)  
di P. Smith e E. Horwood trad. e adatt. di Carlo Panzalis
- 29** Tombola (48K)  
di L. Franzini

### J'soft s.r.l.

#### DIREZIONE, REDAZIONI E AMMINISTRAZIONE

Via Restelli, 5  
20124 Milano  
Tel. (02) 80.841.68.80 842.68.60 843.  
68.68 226.68.37.97

#### DIRETTORE RESPONSABILE:

Pietro Dell'Orco

#### COORDINAMENTO TECNICO:

Luigi Bragagnolo  
Maurizio Cristofari Guzzi

#### REDAZIONE:

Carlo Panzalis  
Franco Serona

#### GRAFICA E IMPAGINAZIONE

Margherita La Noce  
Ivana Rossi  
Raffanella Toffolatti

#### FOTOCOMPOSIZIONE:

Graphotek  
Via Assolani, 18 Milano  
Tel. 02 80.327

#### CONTABILITÀ:

Giulia Pedrazzini  
Flavia Donelli

#### AUTORIZZAZIONE ALLA

**PUBBLICITÀ:**  
Tribunale di Milano  
n° 200 del 14-04-1984

#### STAMPA:

Intergrafica



Picetto (MI)  
Rivista associata  
all'Unione stampa  
Periodica Italiana

#### PUBBLICITÀ

Concessionario per l'Italia e l'Estero  
J'soft s.r.l.

Via Restelli, 5  
20124 MILANO  
Tel. (02)

88.82 895-88.80 606-88.87 233

Tlx 310213 REINA I

Concessionario esclusivo per la  
DIFFUSIONE in Italia e Estero  
SODIP - Via Zuretti, 25  
20125 MILANO

Spedizione in abbonamento  
postale Gruppo II/70  
Prezzo della rivista L. 1.300  
Numero arretrati L. 2.600

**C. TUTTI I DIRITTI DI  
RIPRODUZIONE O TRADUZIONI  
DEGLI ARTICOLI E PUBBLICATI  
SONO RISERVATI**

# STOP BUG

Questo articolo è rivolto esclusivamente ai "Commodoriani" che usano il VIC 20, il C16 o il Plus/4.

Se non l'hai già fatto, memorizza e salva su nastro o disco la versione di STOP BUG per il tuo computer. Questo programma ti permette, una volta caricato ed attivato con RUN, di digitare i listati di Papernsoft senza introdurre errori di battitura.

Come avrai notato, le linee dei nostri programmi terminano con la parola: ":rem" seguita da un numero, detto checksum; **NON COPIARE QUESTA PARTE DELLA LINEA**, che serve solo per tua informazione. Quando STOP BUG è in funzione, infatti, tutte le volte che premi il tasto RETURN dopo aver digitato una linea di programma, appare nell'angolo superiore sinistro dello schermo un numero, che deve coincidere con il checksum da noi pubblicato (sempre che tu non abbia usato comandi abbreviati, come ? per PRINT).

Usando STOP BUG tieni conto del

fatto che gli spazi non influiscono sul checksum, per cui fai particolare attenzione alle istruzioni PRINT dove spesso presenti. Inoltre eventuali inversioni di caratteri non vengono segnalate.

Un altro metodo per utilizzare STOP BUG, soprattutto se sei abituato ad usare le abbreviazioni dei comandi Basic, è controllare i checksum listando il programma e battendo RETURN su ogni linea.

Se intendi sospendere il lavoro di digitazione, puoi salvare su nastro o disco il programma che stai memorizzando senza che STOP BUG interferisca; al momento di riprendere il lavoro ricordati però di caricare ed attivare STOP BUG prima di ricaricare il tuo programma.

Per disabilitare STOP BUG premi RUN/STOP e RESTORE (su VIC 20) oppure RUN/STOP e RESET (su C16 o PLUS/4).

Per riattivarlo batti:

SYS PEEK (56)\*256 per il VIC 20;  
SYS 16128 per il C16 e il Plus/4.

## STOP BUG per C16 e PLUS/4

```
100 POKE55,0:POKE56,63:PRINT"[CLR]ATTENDERE PREGO..."
110 FORI=16128TO16260:READA:CK=CK+A:POKEI,A:NEXT
120 IFCK<>14248THENPRINT"[GIU']ERRORE NELLE ISTRUZIONI DA
    TA":END
130 SYS16128:PRINT"[CLR]{ 2 GIU'}[RVS]STOP BUG{OFF} ATTIV
    ATO.":NEW
200 DATA173,34,3,201,32,208,1,96,141
210 DATA33,63,173,35,3,141,34,63,169
220 DATA32,141,34,3,169,63,141,35,3
230 DATA169,0,133,6,96,32,232,235,133
240 DATA3,134,4,132,5,8,201,13,240
250 DATA17,201,32,240,5,24,101,6,133
260 DATA6,165,3,166,4,164,5,40,96
270 DATA169,13,32,210,255,165,205,141,251
280 DATA3,206,251,3,169,0,133,207,169
290 DATA19,32,210,255,169,18,32,210,255
300 DATA169,58,32,210,255,166,6,169,0
310 DATA133,6,172,33,63,192,232,208,6
320 DATA32,95,164,76,117,63,32,7,208
330 DATA169,32,32,210,255,32,210,255,173
340 DATA251,3,133,205,76,55,63
```

## STOP BUG PER VIC 20

```
100 POKE55,0:POKE56,PEEK(56)-1:R=PEEK(56)*256:PRINT"[CLR]
    ATTENDERE PREGO..."
110 FORI=RTOR+132:READA:CK=CK+A:IFA=999THENA=R/256
120 POKEI,A:NEXT
130 IFCK<>22689THENPRINT"[GIU']ERRORE NELLE ISTRUZIONI DA
    TA":END
140 SYS(R):PRINT"[CLR]{ 2 GIU'}[RVS]STOP BUG{OFF} ATTIVA
    O.":NEW
200 DATA173,36,3,201,32,208,1,96,141
210 DATA33,999,173,37,3,141,34,999,169
220 DATA32,141,36,3,169,999,141,37,3
230 DATA169,0,133,254,96,32,87,241,133
240 DATA251,134,252,132,253,8,201,13,240
250 DATA17,201,32,240,5,24,101,254,133
260 DATA254,165,251,166,252,164,253,40,96
270 DATA169,13,32,210,255,165,214,141,251
280 DATA3,206,251,3,169,0,133,216,169
290 DATA19,32,210,255,169,18,32,210,255
300 DATA169,58,32,210,255,166,254,169,0
310 DATA133,254,172,33,999,192,87,208,6
320 DATA32,205,189,76,117,999,32,205,221
330 DATA169,32,32,210,255,32,210,255,173
340 DATA251,3,133,214,76,55,999
```



## Mini-Mancala

Questo programma trae origine da un antico gioco arabo. Ci sono 4 tazze contrassegnate dalle lettere A,B,C,D: due, la A e la B, appartengono al computer, le altre due, la C e la D, sono tue.

All'inizio del gioco tutte le tazze contengono 2 pedine. A turno i giocatori, cioè tu e il computer, devono prendere le pedine contenute in una delle proprie tazze e distribuire una ad una in senso antiorario nelle altre tazze. Ci dovrà essere sempre almeno una tazza

vuota in quanto le pedine non possono essere rimesse nella tazza da cui sono state prese.

Vince chi riesce a portare tutte le pedine in una delle proprie tazze. Puoi scegliere il livello di difficoltà (da 1 a 3, 1 è il più semplice) e chi deve iniziare per primo. Non scoraggiarti per gli insuccessi iniziali perché il gioco è realmente molto difficile; con un po' di pazienza, tuttavia, riuscirai a mettere a punto una strategia vincente.

```

10 REM *MINI-MANCALA*                                :rem 224
20 COLOR4,2,2:COLOR0,10,3                             :rem 166
30 DIMB(3),SU(3)                                       :rem 91
40 DEF FNSP(X)=15+80*(XAND2)+6*(XAND1)                :rem 89
50 FORI=0TO3                                           :rem 216
60 B(I)=2                                              :rem 177
70 READSU(I)                                          :rem 197
80 NEXT                                              :rem 167
90 DATA2,0,3,1                                       :rem 205
100 ES="{HOME}[ 17 GIU']{ 7 DES}"                     :rem 122
110 GOTO560                                           :rem 102
120 REM *DISEGNO SCHERMO*                             :rem 229
130 PRINT"[CLR][<6>][GIU']"SPC(12)"{RVS} MINI{ 2 *}MANCALA {BLK}" :rem 188
140 PRINT                                              :rem 34
150 PRINTSPC(11)" A"SPC(13)"B"                       :rem 98
160 PRINTSPC(11)"{ 2 SPAZI}[<A>]{ 5 C}[<R>]{ 5 C}[<S>]" :rem 175
170 PRINTSPC(11)"{ 2 SPAZI}-{ 5 SPAZI}-{ 5 SPAZI}-" :rem 153
180 PRINTSPC(11)"{ 2 SPAZI}-{ 2 SPAZI}2{ 2 SPAZI}- :rem 254
    { 2 SPAZI}2{ 2 SPAZI}-"
190 PRINTSPC(11)"{ 2 SPAZI}-{ 5 SPAZI}-{ 5 SPAZI}-" :rem 155
200 PRINTSPC(11)"{ 2 SPAZI}[<Q>]{ 5 C}+{ 5 C}[<W>]" :rem 211

```

210 PRINTSPC(11)"{ 2 SPAZI}-{ 5 SPAZI}-{ 5 SPAZI}-"	:rem	148
220 PRINTSPC(11)"{ 2 SPAZI}-{ 2 SPAZI}2{ 2 SPAZI}-"	:rem	249
230 PRINTSPC(11)"{ 2 SPAZI}-{ 5 SPAZI}-{ 5 SPAZI}-"	:rem	150
240 PRINTSPC(11)"{ 2 SPAZI}[<Z>]{ 5 C}[<E>]{ 5 C}[<X>]"	:rem	185
250 PRINTSPC(11)" C"SPC(13)"D"	:rem	103
260 RETURN	:rem	120
270 REM *IMPUT MOSSA*	:rem	227
280 PRINT\$"MUOVI DALLA TAZZA (C/D){ 2 SPAZI}{ 2 SIN}";	:rem	200
290 INPUTS\$	:rem	162
300 IFSS\$=""THEN280	:rem	227
310 S=-2*(SS\$="C")-3*(SS\$="D")	:rem	80
320 IFS=0ORB(S)=0THEN280	:rem	161
330 RETURN	:rem	118
340 REM *MOSSA COMPUTER*	:rem	193
350 PRINT\$"{ 5 SPAZI}ORA TOCCA A ME{ 7 SPAZI}"	:rem	241
360 FORDE=1TO500:NEXTDE	:rem	177
370 IFLE<3ANDRND(0)*LE<.6THENS=INT(RND(D)*2):GOTO400	:rem	127
380 Q=B(2)+10*(B(1)+10*B(0))	:rem	50
390 S=1	:rem	93
400 IFQ=143ORQ=134ORQ=611ORQ=116THENS=0	:rem	141
410 IFB(S)=0THENS=1-S	:rem	231
420 PRINT\$"{ 3 SPAZI}MUOVO DALLA TAZZA "MID\$("AB",S+1,1)	:rem	156
430 RETURN	:rem	119
440 REM *MOVIMENTO ANTIORARIO*	:rem	134
450 D=S	:rem	109
460 FORC=1TOB(S)	:rem	187
470 FORDE=1TO400:NEXTDE	:rem	178
480 D=SU(D)	:rem	90
490 IFD=STHEN480	:rem	203
500 B(S)=B(S)-1	:rem	252
510 PRINT"{HOME}{ 6 GIU'}"SPC(FNSP(S));B(S)	:rem	19
520 B(D)=B(D)+1	:rem	222
530 PRINT"{HOME}{ 6 GIU'}"SPC(FNSP(D));B(D)	:rem	247
540 NEXTC	:rem	27
550 RETURN	:rem	122
560 REM *PROGRAMMA PRINCIPALE*	:rem	96
570 GOSUB120	:rem	175
580 PRINT\$"{ 4 SPAZI}LIVELLO[ 2 SPAZI](1-3)";	:rem	11
590 INPUTLE	:rem	191
600 IFLE<1ORLE>3THEN580	:rem	146
610 PRINT\$"VUOI INIZIARE TU (S/N)";	:rem	116
620 INPUTAN\$	:rem	219
630 IFAN\$="N"THEN680	:rem	119
640 IFAN\$<>"S"THEN610	:rem	179

```

650 GOSUB270 :rem 180
660 GOSUB440 :rem 180
670 IFB(3)=8THENPRINT$"{ 8 SPAZI}HAI VINTO!{ 8 SPAZI}":G
OTO720 :rem 95
680 GOSUB340 :rem 181
690 GOSUB440 :rem 183
700 IFB(0)=8THENPRINT$"{ 8 SPAZI}HO VINTO!{ 9 SPAZI}":GO
TO720 :rem 27
710 GOTO650 :rem 108
720 PRINT:END :rem 55

```

## Escher

*Questo programma si ispira alle creazioni del famoso artista grafico M.C. Escher. Seguendo le istruzioni che compaiono sullo schermo, utilizzate i caratteri mostrati nelle linee 1000-1200, vale a dire i simboli grafici ottenibili tramite i tasti SHIFT e Commodore, per creare spettacolari disegni. I caratteri vengono raggruppati in "quadrati" di cui è*

*possibile selezionare le dimensioni; i quadrati (e i simboli che li compongono) sono poi orientabili in quattro direzioni (nord-est-sud-ovest) ruotandoli di 90, 180 e 270 gradi rispetto alla posizione iniziale. Vi suggeriamo di fare molte prove per imparare bene l'uso del programma ed ottenere gli effetti desiderati.*

```

10 MK=20 :rem 152
20 PRINT"[CLR]" :rem 198
30 INPUT"COLORE CARATTERI (1-16)";C1:IFC1<1ORC1>16THENPRI
NT"{ 2 SU}":GOTO30 :rem 187
40 INPUT"LUMINANZA CARATTERI (0-7)";C2:IFC2<0ORC2>7THENPR
INT"{ 2 SU}":GOTO40 :rem 73
50 COLOR1,C1,C2 :rem 86
60 INPUT"{GIU'} COLORE SFONDO (1-16)";S1:IFS1<1ORS1>16THEN
PRINT"{ 3 SU}":GOTO60 :rem 189
70 INPUT"LUMINANZA SFONDO (0-7)";S2:IFS2<0ORS2>7THENPRINT
"{ 2 SU}":GOTO70 :rem 169
80 COLOR0,S1,S2 :rem 120
90 INPUT"{GIU'} COLORE BORDO (1-16)";B1:IFB1<1ORB1>16THENP
RINT"{ 3 SU}":GOTO90 :rem 61
100 INPUT"LUMINANZA BORDO (0-7)";B2:IFB2<0ORB2>7THENPRINT
"{ 2 SU}":GOTO100 :rem 119
110 COLOR4,B1,B2 :rem 132

```



120	DIMSY(MK,3)	:rem	97
130	FORCC=0TOMK	:rem	169
140	FORR1=0TO3	:rem	66
150	READSY(CC,R1)	:rem	228
160	NEXT	:rem	214
170	NEXT	:rem	215
180	INPUT("{ 2 GIU'}MISURA DEL QUADRATO (2-23)";S:IFS<2ORS		
	>23THENPRINT("{ 4 SU}":GOTO180	:rem	170
190	SX=INT(39/S)+1	:rem	8
200	SY=INT(23/S)+1	:rem	250
210	DIMDS(SX,SY),BS(S,S),RI(S,S),SC(S*SX,S*SY)	:rem	54
220	PRINT"(CLR)INTRODUCI 1 CARATTERI GRAFICI"	:rem	134
230	PRINT"OGNI RIGA";S;"SIMBOLI{GIU'}"	:rem	227
240	FORR=0TOS-1	:rem	144
250	PRINT"RIGA";R+1;TAB(7);	:rem	73
260	A\$=""	:rem	126
270	INPUTA\$	:rem	142
280	A\$=LEFT\$(A\$+"{ 23 SPAZI}",S)	:rem	47
290	FORK=0TOS-1	:rem	142
300	BS(R,K)=ASC(MID\$(A\$,K+1,1))	:rem	139
310	NEXT	:rem	211
320	NEXT	:rem	212
330	PRINT	:rem	35
340	PRINT"ORIENTAMENTO DEI QUADRATI (N,S,E,O)"	:rem	52
350	PRINT"OGNI FILA";SX;"DIREZIONI"	:rem	196
360	PRINT	:rem	38
370	FORB=0TOSY-1	:rem	221
380	PRINT"FILA";B+1;TAB(7);	:rem	54
390	INPUTA\$	:rem	145
400	FORA=0TOSX-1	:rem	213
410	IN\$=MID\$(A\$,A+1,1):IFIN\$<>"N"ANDIN\$<>"S"ANDIN\$<>"E"AN		
	DIN\$<>"O"THEN850	:rem	142
420	DS(A,B)=-((IN\$="N")+2*(IN\$="E")+3*(IN\$="S")+4*(IN\$="O		
	))-1	:rem	71
430	IFDS(A,B)<0THENDS(A,B)=0	:rem	92
440	NEXT	:rem	215
450	NEXT	:rem	216
460	PRINT	:rem	39
470	PRINT"[ELABORAZIONE DISEGNO]"	:rem	168
480	FORR=0TOS-1	:rem	150
490	FORK=0TOS-1	:rem	144
500	FORR1=3TO0STEP-1	:rem	220
510	FORCC=MKTO0STEP-1	:rem	69
520	IFBS(K,R)=SY(CC,R1) THENC2=CC:R2=R1:CC=0:R1=0:	:rem	128
530	NEXT	:rem	215
540	NEXT	:rem	216
550	BS(K,R)=C2	:rem	251
560	RI(K,R)=R2	:rem	17
570	NEXT	:rem	219
580	NEXT	:rem	220
590	PRINT"[FONDO GRAFICO]"	:rem	152

600	FORR=0TOS-1	:rem	144
610	FORK=0TOS-1	:rem	138
620	FORB=0TOSY-1	:rem	219
630	FORA=0TOSX-1	:rem	218
640	ONDS(A,B)+1GOTO650,660,670,680	:rem	89
650	X=K:Y=R:GOTO690	:rem	175
660	X=S-R-1:Y=K:GOTO690	:rem	142
670	X=S-K-1:Y=S-R-1:GOTO690	:rem	109
680	X=R:Y=S-K-1	:rem	126
690	R1=(DS(A,B)+RI(R,K))AND3	:rem	45
700	SC(X+A*S,Y+B*S)=SY(BS(R,K),R1)	:rem	198
710	NEXT	:rem	215
720	NEXT	:rem	216
730	NEXT	:rem	217
740	NEXT	:rem	218
750	PRINT"[CLR]";	:rem	59
760	FORY=0TO23	:rem	82
770	FORX=0TO39	:rem	89
780	IFSC(X,Y)>255THENPRINT"[RVS]";:GOTO800	:rem	36
790	PRINT"[OFF]";	:rem	62
800	PRINTCHR\$(SC(X,Y)AND255);	:rem	229
810	NEXT	:rem	216
820	NEXT	:rem	217
830	PRINT"[DES]";	:rem	196
840	GOTO840	:rem	113
850	PRINT"[ 2 GIU']INPUT ERRATO!{ 2 SPAZI}RIPROVA.":END	:rem	112
860	DATA32,32,32,32: REM " "	:rem	50
870	DATA165,163,167,164:REM "[<G>]"	:rem	179
880	DATA212,197,217,210:REM "T"	:rem	215
890	DATA199,196,200,198:REM "G"	:rem	223
900	DATA194,195,221,192:REM "B"	:rem	201
910	DATA180,183,170,175:REM "[<H>]"	:rem	184
920	DATA181,184,182,185:REM "[<J>]"	:rem	192
930	DATA161,418,417,162:REM "[<K>]"	:rem	167
940	DATA205,206,205,206:REM "M"	:rem	200
950	DATA203,202,213,201:REM "K"	:rem	187
960	DATA189,173,176,174:REM "[<X>]"	:rem	211
970	DATA207,208,186,204:REM "O"	:rem	215
980	DATA177,171,178,179:REM "[<E>]"	:rem	203
990	DATA169,223,425,479:REM "E"	:rem	193
1000	DATA190,188,172,187:REM "[<V>]"	:rem	244
1010	DATA209,209,209,209:REM "Q"	:rem	255
1020	DATA214,214,214,214:REM "V"	:rem	245
1030	DATA215,215,215,215:REM "W"	:rem	251
1040	DATA219,219,219,219:REM "+"	:rem	16
1050	DATA166,422,166,422:REM "[<+>]"	:rem	214
1060	DATA191,447,191,447:REM "[<B>]"	:rem	250

## Merge

Questo utility permette di sopprimere ad una carenza del Basic del C16 e del Plus/4 per quanto riguarda l'editing dei programmi. Con essa, infatti, potrete caricare un programma Basic senza cancellare quello già presente in memoria (effettuando quindi il "merge", cioè l'unione dei due programmi).

Il programma "merge" va digitato e salvato PRIMA di farlo "girare"; chi utilizza il disk drive tolga la parola REM alla linea 40. La routine si attiva una volta per tutte caricandola e dando il RUN; così facendo un programmino in linguaggio macchina viene memorizzato in un'area di RAM che non interferisce con il Basic. A questo punto, se si desidera "concatenare" due programmi Basic, uno presente in memoria e l'altro registrato su cassetta o disco, è sufficiente digitare

digitate

SYS 1048

ed il merge sarà completato. Fate però attenzione ai "range" dei numeri di linea dei programmi da concatenare: se infatti essi non sono disgiunti possono verificarsi inconvenienti molto spiacevoli. Ad esempio, un programma i cui numeri di linea vanno da 10 a 1000 non può essere concatenato ad un programma il cui range sia 100-2000; prima di effettuare il merge sarà necessario rinumerare uno dei due programmi. Inoltre, se abbiamo un programma A con range 10-500 e un programma B con range 1000-2000, dovremo prima caricare A e poi effettuare il merge con B; infatti, dato che i programmi vengono caricati uno di seguito all'altro, se facessimo il contrario avremmo i numeri di linea "alti" prima di quelli "bassi", e il programma risultante non potrebbe venir eseguito correttamente.

SYS 1024:LOAD "nome programma"

(aggiungendo ",8" se si usa il disk drive). A caricamento effettuato

```

10 PRINT "{CLR}":FOR I=1024 TO 1024+34:READ A:POKE I,A:C=C+A:NEXT I
20 IF C<>3090 THEN PRINT "ERRORE NELLE DATA":END           :rem 199
30 PRINT "PER ESEGUIRE IL MERGE:":PRINT "{GIU'}SYS1024:LOAD  :rem 228
   "CHR$(34)"NOME"CHR$(34);                                :rem 15
40 REM PRINT",8"                                             :rem 125
50 PRINT:PRINT"SYS1048"                                       :rem 201
60 DATA 165,43,141,46,4,165,44,141,47,4,56,165,45,233,2,13  :rem 176
   3,43,165

```

70 DATA 46, 233, 0, 133, 44, 96, 173, 46, 4, 133, 43, 173, 47, 4, 133, 44  
 , 96 : rem 192  
 80 NEW : rem 82

## Merge & Renumber

Queste due utility permettono di sopperire ad alcune carenze del Basic del VIC per quanto riguarda l'editing dei programmi. Con la prima routine, infatti, potrete caricare un programma Basic senza cancellare quello già presente in memoria (effettuando quindi il "merge", cioè l'unione dei due programmi). La seconda, invece, vi offre la possibilità di rinumerare le linee di un programma, aggiornando contemporaneamente gli indirizzi contenuti nelle istruzioni di "salto", vale a dire GOSUB, GOTO, ON... GOTO, ecc.

Il programma "Merge" va digitato e salvato PRIMA di farlo "girare"; chi utilizza il disk drive tolga la paretta REM alla linea 150. La routine si attiva una volta per tutte caricandola e dando il RUN; così facendo un programmino in linguaggio macchina viene memorizzato in fondo alla RAM Basic, modificandone i puntatori. A questo punto, se si desidera "concatenare" due programmi Basic, uno presente in memoria e l'altro registrato su cassetta o disco, è sufficiente digitare

SYS xxxxx "nome programma" (da

nastro)

SYS xxxxx "nome programma",8  
 (da disco)

È lo stesso "Merge" che, appena viene attivato, indica la locazione xxxxx, la quale varia a seconda delle eventuali espansioni di memoria connesse al VIC. Fate però attenzione ai "range" dei numeri di linea dei programmi da concatenare: se infatti essi non sono disgiunti possono verificarsi inconvenienti molto spiacevoli. Ad esempio, un programma i cui numeri di linea vanno da 10 a 1000 non può essere concatenato ad un programma il cui range sia 100-2000; prima di effettuare il merge sarà necessario rinumerare uno dei due programmi. Inoltre, se abbiamo un programma A con range 10-500 e un programma B con range 1000-2000, dovremo prima caricare A e poi effettuare il merge con B; infatti, dato che i programmi vengono caricati uno di seguito all'altro, se facessimo il contrario avremmo i numeri di linea "alti" prima di quelli "bassi", e il programma risultante non potrebbe venir eseguito correttamente. Il programma "Renumber" è intera-

mente scritto in Basic e numerato a partite dalla linea 63000, in modo da poter coesistere praticamente con qualsiasi altro programma Basic (infatti in genere non vengono mai usati numeri di linea così alti). Il programma da rinumerare, una volta caricato in memoria, deve venir sottoposto ad un piccolo "adattamento", che consiste nel modificare gli indirizzi delle istruzioni di "salto" in modo che essi siano sempre composti da 5 cifre (esempio: GOTO 20 deve essere sostituito con GOTO 00020). "Renumber" va poi concatenato tramite un merge al programma da rinumerare, ed

attivato digitando RUN 63000. Eventuali errori commessi nella correzione degli indirizzi vengono segnalati bloccando l'elaborazione; dopo aver provveduto alla correzione, digitare nuovamente RUN 63000. Quando la rinumerazione è completa appare il messaggio "FATTO"; abbiate un po' di pazienza perché il procedimento può richiedere anche diversi minuti. I numeri di linea partono ora da 10000 ed hanno incrementato 10; il programma rinumerato può quindi venir salvato dopo aver cancellato manualmente le linee del "Renumber" (63000-63360).

## Merge

```

100 POKE55,0:POKE56,PEEK(56)-1:CLR:PRINT"{CLR}" :rem 126
110 S=PEEK(56)*256+1:FORJ=STOS+78:READV :rem 93
120 C=C+V:POKEJ,V:NEXT :rem 133
130 IF C<>8785 THEN PRINT "ERRORE NELLE DATA":END :rem 38
140 PRINT "PER EFFETTUARE{ 8 SPAZI} IL MERGE:":PRINT "{GIU'}
    SYS"S;CHR$(34)"NOME"CHR$(34); :rem 64
150 REM PRINT",8" :rem 175
160 DATA169,0,133,10,32,209,225,165 :rem 226
170 DATA43,72,165,44,72,56,165,45 :rem 147
180 DATA233,2,133,43,165,46,233,0 :rem 125
190 DATA133,44,169,0,133,185,166,43 :rem 240
200 DATA164,44,169,0,32,213,255,176 :rem 232
210 DATA14,134,45,132,46,32,51,197 :rem 179
220 DATA104,133,44,104,133,43,96,170 :rem 16
230 DATA201,4,144,244,240,10,104,133 :rem 0
240 DATA44,104,133,43,24,108,0,3 :rem 65
250 DATA164,186,136,240,209,208,239 :rem 246
260 PRINT:NEW :rem 73

```

## Renumber

```

63000 CLR:DIMZZ(FRE(0)/12,1):IN=PEEK(44)*256+PEEK(43):LA=
    IN:PP=LA :rem 58
63010 DEFFNDH(X)=PEEK(X)+256*PEEK(X+1) :rem 185
63020 DEFFNH1(X)=XAND255 :rem 132
63030 DEFFNH2(X)=INT(X/256) :rem 31
63040 IF PP<>FNDH(LA) THEN 63080 :rem 212
63050 LA=FNDH(LA):NL=FNDH(LA+2):IF NL=63000 THEN 63270
    :rem 252

```



```

63060 IFPEEK(LA+5)=143THENPP=FNDH(LA):GOTO63050 :rem 65
63070 PP=PP+4 :rem 220
63080 IFPEEK(PP)<>167THEN63120 :rem 233
63090 S=0:IFPEEK(PP+1)=32THENS=1 :rem 143
63100 IFPEEK(PP+1+S)<48ORPEEK(PP+1+S)>57THEN63260 :rem 140
63110 GOTO63130 :rem 49
63120 IFPEEK(PP)<>137ANDPEEK(PP)<>141THEN63260 :rem 223
63130 S=0:IFPEEK(PP+1)=62THENS=1 :rem 141
63140 GG$="":FORI=1+STO5+S:IFPEEK(PP+I)<48ORPEEK(PP+I)>57
THEN63250 :rem 220
63150 GG$=GG$+CHR$(PEEK(PP+I)):NEXT :rem 32
63160 GG=VAL(GG$):L1=IN:L2=FNDH(IN+2):LL=10000 :rem 150
63170 IFL2=63000THENPRINT"LINEA NON DEFINITA{ 4 SPAZI}ALL
A LINEA";NL:END :rem 69
63180 IFL2=GGTHEN63220 :rem 6
63190 L1=FNDH(L1):L2=FNDH(L1+2):LL=LL+10 :rem 30
63200 IFPEEK(L1+4)=143ANDPEEK(L1+5)=35THENLL=1000*INT((LL
+1000)/1000) :rem 108
63210 GOTO63170 :rem 54
63220 ZZ(ZI,0)=LL:ZZ(ZI,1)=PP+S:ZI=ZI+1 :rem 137
63230 IFPEEK(PP+S+6)=44THENPP=PP+6+S:GOTO63130 :rem 34
63240 GOTO63260 :rem 57
63250 PRINT"COMANDO NON STANDARD{ 2 SPAZI}ALLA LINEA";NL:
END :rem 119
63260 PP=PP+1:GOTO63040 :rem 74
63270 LA=IN:LL=10000 :rem 99
63280 IFPEEK(LA+4)=143ANDPEEK(LA+5)=35THENLL=1000*INT((LL
+1000)/1000) :rem 148
63290 IFFNDH(LA+2)=63000THEN63320 :rem 81
63300 POKELA+2,FNH1(LL):POKELA+3,FNH2(LL) :rem 174
63310 LL=LL+10:LA=FNDH(LA):GOTO63280 :rem 110
63320 IFZI=0THEN63360 :rem 206
63330 FORI=0TOZI-1 :rem 64
63340 FORJ=1TO5:POKEZZ(I,1)+J,ASC(MID$(STR$(ZZ(I,0)),J+1)
):NEXT :rem 6
63350 NEXT :rem 64
63360 PRINT"FATTO":END :rem 98

```

W  
E  
B

**Fuoristrada**

3K RAM - joystick

È stata inventata una rivoluzionaria | superficie lunare, che è piena zep-  
auto fuoristrada per esplorare la | pa di crateri che rendono difficile

Avete a disposizione una scorta limitata di carburante (999 litri) che

Ed ora buon divertimento!!

15

```

370 XS="{ 3 SU}{ 2 SIN}{YEL}!#{WHT}$ {GIU'}{DES}{GRN}%&'
    {GIU'}{DES}{ 3 SPAZI}" :rem 123
380 YS="{ 3 SU}{ 2 SIN}{YEL}{ 3 SPAZI}":ZS="{ 2 SU}
    { 2 DES}{YEL}!#{WHT}$ {GIU'}{DES}{GRN}%&'" :rem 130
390 PS="{ 2 SU}{ 2 DES}{YEL}!#{WHT}$ {GIU'}{DES}{GRN}}&,"
    :rem 191
400 S1=36874:S2=S1+1:S3=S2+1:S4=S3+1:V=S4+1 :rem 171
410 REM CREA L'ORIZZONTE :rem 219
420 PRINT"{HOME}{ 4 GIU'}{ 4 SIN}{GRN}{WHT}."{ 4 DES}."SPC
    (12)".{DES}"; :rem 137
430 PRINT"{ 2 DES}."{ 5 DES}{GRN}{RVS}NM{ 2 DES}NM{OFF}
    { 5 DES}{WHT}."{ 6 DES}"; :rem 64
440 PRINT"{RVS}{ 7 DES}{GRN}N{ 2 DES}MN{ 2 DES}M{OFF}
    {WHT}."{DES}{RVS}{GRN}NM{OFF}{ 3 DES}{WHT}."{ 3 DES}";
    :rem 105
450 PRINT"{RVS}{ 2 DES}{GRN}NM{ 2 DES}N{ 8 DES}MN
    { 2 DES}M{OFF}{DES}{WHT}."{ 2 DES}."{DES}"; :rem 163
460 PRINT"{RVS}{DES}{GRN}N{ 2 SPAZI}MN"SPC(14)"M{ 2 DES}
    NMN"; :rem 172
470 PRINT"{RVS}N"SPC(20)"MN{ 3 DES}" :rem 27
480 REM VISUALIZZA IL PUNTEGGIO :rem 217
490 PRINTCHR$(19)CHR$(158)"DISTANZA:"DI:PRINTCHR$(19)SPC(
    15)"CARB:"FU :rem 144
500 REM VISUALIZZA LA MACCHININA :rem 227
510 PRINTCHR$(159)S$;RIGHT$(Q$,26) :rem 194
520 PRINTS$"{ 2 SU}{ 2 DES}{YEL}!#{WHT}$ {GIU'}{DES}{GRN}%
    &'" :rem 4
530 FORL=1TO1000:NEXT :rem 22
540 REM RUMORE DEL MOTORE :rem 2
550 POKEV,5:POKES1,128:POKES4,254 :rem 235
560 REM ROUTINE PRINCIPALE :rem 140
570 FORT=NT03:IFPEEK(PO(T)+Y)=43THEN710 :rem 170
580 NEXTT :rem 48
590 IFDI=250THENFU=FU+500 :rem 240
600 IFZ<>NTHENIFPEEK(841)=OTHENZ=N:GOSUB830 :rem 13
610 IFZ=NTHENZ1=Z1+N:IFZ1=4THENZ1=0:Z=0:FORT=NT03:PO(T)=P
    O(T)+Y:NEXT :rem 3
620 IFZ=NTHENFU=FU-5 :rem 16
630 PRINTCHR$(19)CHR$(5)SPC(9);DI:PRINTCHR$(19)SPC(20);IN
    T(FU)" " :rem 19
640 G$=RIGHT$(Q$,X)+LEFT$(Q$,N):Q$=G$:PRINTCHR$(159)S$;RI
    GHT$(Q$,Y) :rem 150
650 IFZ=NTHENPRINTS$;X$ :rem 249
660 IFZ=OTHENPRINTS$;Y$:IFRND(1)<.5THENPRINTS$;Z$:GOTO680
    :rem 199
670 IFZ=OTHENPRINTS$;P$ :rem 244
680 FU=FU-.2:IFFU<OTHEN710 :rem 84
690 DI=DI+N :rem 111
700 GOTO570 :rem 108
710 REM CRASH :rem 237
720 POKES1,0 :rem 168

```

```

730 FORVO=15TO0STEP-.5:R=INT(RND(1)*9+195):POKE(PO(1)),R:
    POKE(PO(2)),R:POKE(PO(3)),R                                :rem 14
740 POKE36875,44+INT(RND(1)*3+1)                                :rem 241
750 POKES3,INT(RND(1)*5+249)                                    :rem 72
760 POKEV,VO:POKES4,160+VO:NEXTVO:POKES4,0:POKES3,0          :rem 43
                                                                    :rem 248
770 POKE646,INT(RND(1)*7+1)                                      :rem 248
780 PRINT$"{ 2 SU}{ 2 SIN}{RVS}Z{OFF} FUORISTRADA {RVS}
    Z{OFF}{GIU'}{ 6 SIN}{ 3 SPAZI}"                            :rem 66
790 PRINTLEFT$(S$,4)+RIGHT$(S$,15)"-([ 3 SPAZI]PREMI
    { 2 SPAZI}FUOCO{ 3 SPAZI})-"                                :rem 98
800 IFPEEK(845)<>0THEN770                                        :rem 181
810 PRINTCHR$(147):GOTO310                                      :rem 27
820 REM PDSIZIONE DELLA MACCHININA                             :rem 107
830 FORT=1TO3:PO(T)=PO(T)-26:NEXT                              :rem 39
840 RETURN                                                       :rem 124
850 REM PAGINA DI PRESENTAZIONE                                :rem 165
860 PRINTCHR$(147):POKE36879,8                                  :rem 247
870 PRINTCHR$(5)"{ 3 GIU'}{ 4 DES}* FUORISTRADA *"          :rem 54
880 PRINT"{ 3 GIU'}{ 6 DES}*PAPERSOFT*"                      :rem 90
890 RETURN                                                       :rem 129
900 REM **ALFABETO** (BLOCCO DATI 1)                          :rem 185
910 DATA0,0,0,0,0,0,0,0                                       :rem 104
920 DATA252,132,132,254,134,134,134,0                       :rem 65
930 DATA252,132,132,254,134,134,254,0                       :rem 69
940 DATA252,128,128,128,128,128,254,0                       :rem 86
950 DATA240,136,132,134,134,134,254,0                       :rem 69
960 DATA248,128,128,252,128,128,254,0                       :rem 91
970 DATA254,128,128,252,128,128,128,0                       :rem 89
980 DATA252,128,128,134,134,134,254,0                       :rem 81
990 DATA132,132,132,254,134,134,134,0                       :rem 69
1000 DATA16,16,16,56,56,56,56,0                             :rem 32
1010 DATA254,16,16,24,24,24,248,0                            :rem 121
1020 DATA128,132,136,240,140,140,140,0                      :rem 97
1030 DATA128,128,128,128,128,128,254,0                      :rem 127
1040 DATA204,180,132,134,134,134,134,0                      :rem 104
1050 DATA132,132,228,158,134,134,134,0                      :rem 114
1060 DATA252,132,132,134,134,134,254,0                      :rem 109
1070 DATA254,130,254,128,128,128,128,0                      :rem 124
1080 DATA252,132,132,150,142,134,254,1                      :rem 109
1090 DATA254,130,252,134,134,134,134,0                      :rem 112
1100 DATA248,128,252,6,6,6,254,0                             :rem 79
1110 DATA254,16,16,24,24,24,24,0                             :rem 66
1120 DATA132,132,132,134,134,134,254,0                      :rem 103
1130 DATA132,132,196,70,102,38,62,0                         :rem 220
1140 DATA132,132,132,150,150,150,254,0                      :rem 99
1150 DATA68,68,40,16,40,68,68,0                              :rem 44
1160 DATA132,68,52,14,6,6,6,0                                :rem 184
1170 DATA126,2,30,32,64,128,254,0                            :rem 122.
1180 REM CHECKSUM                                              :rem 1

```

1190 DATA24607	:rem	232
1200 REM **DATI CARATTERI** (BLOCCO DATI 2)	:rem	103
1210 DATA128,64,33,17,19,63,112,207	:rem	229
1220 DATA0,0,0,0,0,0,0,0	:rem	147
1230 DATA32,252,194,217,220,255,124,187	:rem	177
1240 DATA0,0,0,0,128,248,28,230	:rem	13
1250 DATA223,185,182,118,121,63,31,15	:rem	71
1260 DATA215,215,238,238,239,199,199,1	:rem	150
1270 DATA247,59,218,220,60,248,248,224	:rem	140
1280 DATA53,86,136,0,0,0,0,0	:rem	121
1290 DATA223,185,176,112,121,63,31,15	:rem	72
1300 DATA78,178,1,0,0,0,0,0	:rem	66
1310 DATA128,129,129,129,198,198,68,56	:rem	159
1320 DATA247,59,26,28,60,248,248,224	:rem	43
1330 REM CHECKSUM	:rem	254
1340 DATA10397	:rem	230
1350 REM **L.M. JOYSTICK** (BLOCCO DATI 3)	:rem	18
1360 DATA120,169,78,141,20,3,169,3	:rem	184
1370 DATA141,21,3,88,96,0,0,0,0,0	:rem	108
1380 DATA169,0,141,19,145,169,127	:rem	146
1390 DATA141,34,145,173,17,145,41	:rem	136
1400 DATA4,141,73,3,173,17,145,41	:rem	124
1410 DATA8,141,74,3,173,17,145,41	:rem	130
1420 DATA16,141,75,3,173,32,145,41	:rem	176
1430 DATA128,141,76,3,173,17,145	:rem	88
1440 DATA41,32,141,77,3,169,255	:rem	40
1450 DATA141,34,145,169,128,141	:rem	40
1460 DATA19,145,76,191,234,0,0,0,0	:rem	170
1470 REM CHECKSUM	:rem	3
1480 DATA7167	:rem	188

Questo programma didattico simula l'utilizzo di un calibro a coulisse, e può essere molto utile per chi è ancora alle prime armi con le misurazioni. Presenta anche la grafica, e la seconda opzione è trasformata in un piacevole giochino. Verrà stampata la figura del calibro con l'unità di misura su di essa, ed il nonio sottó: a seconda dell'op-

zione che avrete scelto, potrete o posizionare il nonio su una misura che volete, oppure potrete rilevare voi stessi la misura (si spera senza errori).

Si consiglia di iniziare con la prima opzione, e farla seguire dalla seconda; dopodiché (si spera), avrete imparato ad usare un calibro a coulisse.



```

100 REM
110 CALL CLEAR
120 CALL HCHAR(23,10,42,17)
130 PRINT TAB(8);"*"
140 PRINT TAB(8);"* C A L I B R O *"
150 PRINT TAB(8);"*"
160 CALL HCHAR(24,10,42,17)
170 PRINT
180 PRINT
190 PRINT " Il programma simula l'utilizzo d'un calibro
    coulisse munito di un vernier a 1/10."
200 PRINT
210 PRINT " Desiderate :"
220 PRINT "- posizionare seguendo una data coordinata
    ? Premi 1."
230 PRINT
240 PRINT "- vuoi esercitarti a misurare il calibro ten-
    nando ad azzardo ? Premi 2."
250 PRINT
260 PRINT
270 PRINT "Utilizza il PUNTO DECIMALE e non la virgola (es-
    empio: 3.4e non 3,4)"
280 PRINT
290 PRINT " Premi un tasto qualunque per iniziare e pas-
    sare a una scelta."
300 PRINT
310 INPUT "La tua scelta ? ":R
320 IF R=1 THEN 390
330 CALL CLEAR
340 INPUT "QUAL'E' IL VOSTRO NOME ? ":NOM$
350 PRINT
360 INPUT " Quante misure vuoi fare ":NM
370 PRINT
380 PRINT " se vuoi continuare, premi un tasto."
390 CALL KEY(0,KE,ST)
400 IF ST=0 THEN 390
410 CALL CLEAR
420 CALL SCREEN(5)
430 CALL COLOR(2,2,15)
440 CALL COLOR(3,2,15)
450 CALL COLOR(4,2,15)
460 CALL COLOR(5,2,15)
470 CALL COLOR(6,2,15)
480 CALL COLOR(7,2,15)
490 CALL COLOR(8,2,15)
500 CALL COLOR(9,2,5)
510 CALL COLOR(10,2,5)
520 CALL COLOR(11,2,5)
530 CALL COLOR(15,2,15)
540 CALL COLOR(13,2,15)
550 CALL COLOR(14,5,15)

```

```

560 CALL COLOR(16,2,15)
570 CALL COLOR(12,7,15)
580 CALL CHAR(149,"0101010101010101")
590 CALL CHAR(150,"")
600 CALL CHAR(128,"80808080808080")
610 CALL CHAR(129,"40404040404040")
620 CALL CHAR(130,"20202020202020")
630 CALL CHAR(131,"10101010101010")
640 CALL CHAR(132,"08080808080808")
650 CALL CHAR(133,"04040404040404")
660 CALL CHAR(134,"02020202020202")
670 CALL CHAR(135,"01010101010101")
680 CALL CHAR(136,"")
690 CALL CHAR(144,"80808080808080FF")
700 CALL CHAR(145,"20202020202020FF")
710 CALL CHAR(146,"08080808080808FF")
720 CALL CHAR(147,"02020202020202FF")
730 CALL CHAR(148,"00000000000000FF")
740 CALL CHAR(152,"60909090909060")
750 CALL CHAR(153,"E08080C02020C0")
760 CALL CHAR(154,"26292929292926")
770 CALL CHAR(155,"27242426212126")
780 CALL CHAR(137,"F0F0F0F0F0F0F0F0")
790 CALL CHAR(123,"F0F0F0F0F0F0F0F0")
800 CALL CHAR(126,"FFFFFFFFFFFFFFFF")
810 IF R=2 THEN 1070
820 CALL CLEAR
830 GOSUB 2610
840 REM
850 MOT$=CHR$(150)&"LATO"&CHR$(150)&"?"&CHR$(150)
860 D=2
870 LI=20
880 GOSUB 3230
890 CALL SOUND(100,-2,5)
900 GOSUB 2350
910 GM=GE
920 IF GM<11.1 THEN 1010
930 MOT$="VALORE"&CHR$(150)&"TROPP0"&CHR$(150)&"GRANDE"
940 LI=22
950 O=2
960 GOSUB 3230
970 FOR V=1 TO 4
980 CALL SOUND(200,-7,5)
990 NEXT V
1000 GOTO 1020
1010 GOSUB 2820
1020 CALL KEY(0,K,E)
1030 IF E=0 THEN 1020
1040 GOSUB 3140
1050 GOTO 840
1060 REM

```

```

1070 CALL CLEAR
1080 NBE=0
1090 BR=0
1100 GOSUB 2610
1110 FOR CP=1 TO NM
1120 RANDOMIZE
1130 MR=0
1140 GM=INT(RND*110)/10
1150 NBE=NBE+1
1160 GOSUB 2820
1170 MOT$=CHR$(150) & "LATO" & CHR$(150) & "?" & CHR$(150)
1180 LI=20
1190 D=1
1200 GOSUB 3230
1210 CALL SOUND(100,-2,5)
1220 FOR I=12 TO 16
1230 CALL HCHAR(I,26,150,7)
1240 NEXT I
1250 MOT$="SCORE"
1260 LI=13
1270 D=26
1280 GOSUB 3230
1290 GOSUB 2350
1300 IF GE=GM THEN 1730
1310 IF MR=0 THEN 1340
1320 IF MR=1 THEN 1480
1330 IF MR=2 THEN 1670
1340 MOT$="ERRORE:PROVATE"&CHR$(150) & "ANCORA"
1350 LI=22
1360 D=2
1370 FOR I=1 TO 4
1380 CALL SOUND(90,165,5)
1390 CALL SOUND(90,131,5)
1400 CALL SOUND(90,165,5)
1410 CALL SOUND(200,165,30)
1420 NEXT I
1430 GOSUB 3230
1440 FOR DEL=1 TO 200
1450 NEXT DEL
1460 GOSUB 3180
1470 GOTO 1650
1480 IF INT(GE)=INT(GM) THEN 1520
1490 IF GE-INT(GE)=GM-INT(GM) THEN 1540
1500 MOT$="errore totale !"
1510 GOTO 1550
1520 MOT$="errore nella parte decimale"
1530 GOTO 1550
1540 MOT$="errore nella parte intera"
1550 LI=22
1560 D=2
1570 GOSUB 3230

```

```

1580 MOT$="ULTIMA"&CHR$(150)&"PROVA"
1590 LI=23
1600 D=2
1610 GOSUB 3230
1620 FOR DEL=1 TO 400
1630 NEXT DEL
1640 GOSUB 3180
1650 MR=MR+1
1660 GOTO 1170
1670 MOT$="ERRORE:IL"&CHR$(150)&"LATO"&CHR$(150)&"E' "&CHR
$(150)&STR$(GM)
1680 FOR V=1 TO 5
1690 CALL SOUND(50,110,5)
1700 CALL SOUND(100,-5,5)
1710 NEXT V
1720 GOTO 1790
1730 MOT$="BENE"
1740 BR=BR+1
1750 FOR V=1 TO 5
1760 CALL SOUND(50,440,5)
1770 CALL SOUND(50,660,5)
1780 NEXT V
1790 LI=22
1800 D=3
1810 GOSUB 3230
1820 MOT$=STR$(BR)&"/"&STR$(NBE)
1830 LI=15
1840 D=26
1850 GOSUB 3230
1860 MOT$="premi un tasto qualunque"
1870 LI=24
1880 D=2
1890 GOSUB 3230
1900 CALL KEY(0,CL,ET)
1910 IF ET=0 THEN 1900
1920 GOSUB 3140
1930 NEXT CP
1940 CALL CLEAR
1950 PRINT
1960 PRINT
1970 CALL SCREEN(4)
1980 FOR I=2 TO 16
1990 CALL COLOR(I,2,4)
2000 NEXT I
2010 PRINT TAB(10);NOM$;" ,"
2020 PRINT
2030 PRINT "      Avete indovinato"
2040 IF BR>1 THEN 2070
2050 PRINT TAB(7);STR$(BR)&" misura su "&STR$(NM)
2060 GOTO 2080
2070 PRINT TAB(7);STR$(BR)&" misure su "&STR$(NM)

```

```

2080 PRINT
2090 IF BR/NM>.8 THEN 2220
2100 IF BR/NM>.5 THEN 2180
2110 PRINT " Fai proprio pena !"
2120 PRINT
2130 PRINT " Rifai ancora questa prova ! Riguardat
i di piu' la posizione ZERO DEL CUR- SORE"
2140 PRINT "cne da la PARTE INTERA dellamisura,"
2150 PRINT "e la gradazione del cursore che COINCIDE ha u
n tratto diparte fissa"
2160 PRINT "(questa gradazione ti da' laPARTE DECIMALE de
lla misura)"
2170 GOTO 2250
2180 PRINT " Non male..."
2190 PRINT
2200 PRINT "ma poteva essere un 10 in pagella. Peccato!
Comunque, riprovaci un'altra volta."
2210 GOTO 2250
2220 PRINT " Molto bene !"
2230 PRINT
2240 PRINT "Ora potete dire che sapete usare un calibro
a 1/10"
2250 PRINT
2260 INPUT "Un'altra serie ? ":REP$
2270 IF (REP$="SI")+(REP$="S") THEN 340
2280 CALL CLEAR
2290 PRINT
2300 PRINT
2310 PRINT TAB(7);"ARRIVEDERCI"
2320 PRINT
2330 PRINT
2340 END
2350 DIM NB(5)
2360 FOR I=1 TO 5
2370 IF K=13 THEN 2440
2380 CALL KEY(0,K,E)
2390 IF E=0 THEN 2380
2400 IF E=-1 THEN 2380
2410 IF K=13 THEN 2430
2420 CALL HCHAR(20,10+I,K)
2430 NB(I)=K
2440 NEXT I
2450 K=14
2460 IF NB(1)<>46 THEN 2490
2470 GE=(NB(2)-48)/10
2480 GOTO 2600
2490 IF NB(2)<>46 THEN 2520
2500 GE=NB(1)-48+(NB(3)-48)/10
2510 GOTO 2600
2520 IF NB(3)<>46 THEN 2550
2530 GE=(NB(1)-48)*10+NB(2)-48+(NB(4)-48)/10

```



```

2540 GOTO 2600
2550 IF NB(2)<>13 THEN 2580
2560 GE=NB(1)-48
2570 GOTO 2600
2580 IF NB(3)<>13 THEN 2600
2590 GE=(NB(1)-48)*10+NB(2)-48
2600 RETURN
2610 REM
2620 CALL HCHAR(2,3,150,29)
2630 CALL HCHAR(3,3,150,29)
2640 CALL HCHAR(4,3,150,29)
2650 CALL HCHAR(4,5,148,2)
2660 CALL VCHAR(5,3,150,10)
2670 CALL VCHAR(5,4,149,12)
2680 FOR I=0 TO 24
2690 PH=I-INT(I/5)*5
2700 CPH=144+PH
2710 C=7+I
2720 CALL HCHAR(4,C,CPH)
2730 NEXT I
2740 CALL HCHAR(3,7,152)
2750 CALL HCHAR(3,13,153)
2760 CALL HCHAR(3,19,154)
2770 CALL HCHAR(3,25,155)
2780 GOSUB 3140
2790 CALL SOUND(100,660,5)
2800 RETURN
2810 REM
2820 T=GM*10+1
2830 C=INT(T/8)+7
2840 IF (GM>.6)+(GM=0) THEN 2890
2850 CALL VCHAR(5,5,137,2)
2860 CALL VCHAR(7,5,123,3)
2870 CALL VCHAR(10,5,137,7)
2880 GOTO 2930
2890 FOR Q=5 TO C-3
2900 CALL VCHAR(7,Q,126,3)
2910 NEXT Q
2920 CALL VCHAR(5,C-2,150,12)
2930 CALL VCHAR(7,C-1,150,8)
2940 CALL HCHAR(5,C-1,150,15)
2950 CALL HCHAR(6,C-1,150,15)
2960 P=T-INT(T/8)*8
2970 FOR I=P TO P+11
2980 PP=I-INT(I/9)*9
2990 IF I=0 THEN 3010
3000 IF I<>9 THEN 3030
3010 PP=9
3020 CALL HCHAR(5,C-1,135)
3030 IF I<>18 THEN 3060
3040 CALL HCHAR(5,C,135)

```

```

3050 GOTO 3070
3060 CALL HCHAR(5,C,127+PP)
3070 C=C+1
3080 NEXT I
3090 CALL HCHAR(6,C-1,154)
3100 CALL HCHAR(6,C-6,153)
3110 CALL HCHAR(6,C-12,152)
3120 CALL SOUND(100,-3,5)
3130 RETURN
3140 REM
3150 FOR Q=5 TO 32
3160 CALL VCHAR(5,Q,32,12)
3170 NEXT Q
3180 FOR M=19 TO 24
3190 CALL HCHAR(M,1,136,32)
3200 NEXT M
3210 RETURN
3220 REM
3230 FOR I=1 TO LEN(MOTS)
3240 L$=SEG$(MOTS,I,1)
3250 CALL HCHAR(LI,I+D,ASC(L$))
3260 NEXT I
3270 RETURN

```

## Missione impossibile

16 48K

*Aggiratevi nel tunnel sul pianeta alieno, bombardando le postazioni nemiche ed evitando di essere colpiti dalle astronavi extraterrestri. Il gioco vi richiede abilità e tempismo: dovrete infatti riuscire a sganciare le bombe in modo che esse intercettino la rotta delle navi*

*nemiche. Infine tenete conto che il programma è rivolto solo ad utenti dotati di spiccato senso dell'umorismo: la missione è veramente impossibile!*

*Chi ha una Issue 3 sostituisca alla linea 1200 il numero 255 con il numero 191.*

```

1 REM *****
2 REM  MISSIONE  IMPOSSIBILE
3 REM  *****
5 IF PEEK USR "a"=128 THEN GO TO 290
10 FOR a=1 TO 13: READ a$
20 FOR b=0 TO 7: READ c
30 POKE USR a$+b,c
40 NEXT b

```

```

50 NEXT a
60 DATA "a",128,64,32,16,8,4,2,1
70 DATA "b",1,2,4,8,16,32,64,128
80 DATA "c",0,0,0,0,3,12,48,192
90 DATA "d",3,12,48,192,0,0,0,0
100 DATA "e",192,48,12,3,0,0,0,0
110 DATA "f",0,0,0,0,192,48,12,3
120 DATA "g",0,16,16,40,40,84,108,0
130 DATA "h",255,16,184,230,184,16,255,0
140 DATA "i",146,84,0,198,0,84,146,0
150 DATA "j",0,0,3,14,3,0,0,0
160 DATA "k",255,0,0,0,0,0,0,0
170 DATA "l",0,126,126,126,126,126,126,0
180 DATA "m",0,0,6,26,100,26,6,0
290 BORDER 0: PAPER 0: INK 7: CLS
300 LET a$="BKKKKEFBKACDKKEFBKKKACDKKKKABABA"
310 LET b$="{G8}GGGG{3G8}G{3G8}GG{3G8}GGG{3G8}GGGG{5G8}"
320 LET c$=" CDA BABA COEF BEF CDA CDA CDA"
340 LET f:=0: LET sc=0
370 LET fu=32
380 LET li=2
700 PLOT 0,15: DRAW 255,0
710 PLOT 0,96: DRAW 255,0
720 PRINT AT 8,1:"PUNTI:";sc
730 GO SUB 9500
760 INPUT AT 0,0:"LIVELLO DI DIFFICOLTA' (1-5)? { 1 = FACILE / 5 =
    DIFFICILE } ";v
761 IF v>5 OR v<1 THEN GO TO 760
765 LET v=v-1
767 PRINT AT 0,1: INK 7:"MISSIONE IMPOSSIBILE"
770 BEEP 2,30
780 PRINT AT 21,0;" "
790 FOR z=6 TO 2 STEP -1
800 PRINT INK z;AT z,0:"LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL"
810 NEXT z
815 POKE 23658,0
820 PRINT AT 12,1:"TASTI - P - AVANTI";AT 13,9:"O - ARRESTA";AT 14,9
    ;"Q - ALTO";AT 15,9:"A - BASSO";AT 16,9:"M - BOMBA"
830 PRINT AT 21,3:"PREMI ";FLASH 1;"BOMBA"; FLASH 0;" PER GIOCARE"
840 IF INKEY$<>"m" THEN GO TO 840
850 PRINT AT 21,0;,,
900 FOR z=11 TO 17
910 PRINT AT z,0;" "
920 NEXT z
930 LET a=12: LET b=8
940 LET c=0: LET d=2: LET e=2
950 LET g=0: LET h=0: LET i=0
960 LET j=0: LET k=10: LET l=10
990 POKE 23658,0
1000 LET z$=INKEY$
1010 LET a1=a: LET b1=b
1020 IF z$="p" THEN LET b=b+2
1030 LET b=b-1
1040 LET a=a+(z$="a" AND a<17)-(z$="q" AND a>11)
1045 IF z$="o" THEN LET b=b+1
1048 IF b>31 THEN GO SUB 6000
1060 PRINT AT a1,b1;" "
1065 IF b<0 THEN GO TO 4000
1070 PRINT INK 4;AT a,b;"H"
1080 IF b1>b AND a1=a THEN GO TO 1100
1090 IF b1>0 THEN PRINT INK 2;AT a1,b1-1;" "

```

```

1095 IF b>0 THEN PRINT INK 2;AT a,b-1;"J"
1100 GO SUB 9000: GO SUB 9100
1106 IF a=INT k AND l=b THEN GO TO 4000
1190 LET z$=INKEY$
1200 IF c<>1 AND IN 32766=255 THEN GO TO 1300
1210 IF c=0 THEN LET d=a: LET e=b: LET c=1: LET fu=fu-1: PRINT AT li
, fu;" ": IF fu=0 THEN GO TO 4000
1220 LET d1=d: LET e1=e
1230 LET d=d+1: LET e=e+1
1240 PRINT AT d1,e1;" "
1250 PRINT INK 6;AT d,e;"."
1260 IF e>30 THEN PRINT AT d,e;" ": LET c=0
1270 IF d>17 THEN GO TO 2000
1280 IF i=e AND d=h THEN GO TO 3000
1290 IF INT k=d AND l=e THEN GO TO 3500
1300 IF f<14 THEN GO TO 1400
1310 IF a$<>"BKKKKKEFBKACDKKEFBKKKACDKKKKABABA" THEN GO TO 1400
1320 LET b$="{G8}GGGG{3G8}G{3G8}GG{3G8}GGG{3G8}GGGG{5G8}"
1330 LET f=0
1400 IF g=0 AND (b+(18-a)-v-2<1 OR b+(18-a)-v-1>32) THEN GO TO 1500
1410 IF g=0 AND (" "+b$+" ")(b+(18-a)+v+3)<>"G" THEN GO TO 1
500
1420 IF g=1 THEN GO TO 1450
1430 LET g=1: LET i=b+(18-a)-v-2: LET h=18
1440 LET b$(b+(18-a)-v-1)=" ": LET f=f+1
1450 LET h1=h: LET i1=i
1460 IF b<i THEN LET i=i-1
1470 IF b>i THEN LET i=i+1
1475 LET h=h-1
1480 PRINT AT h1,i1;" "
1483 IF h=10 THEN LET g=0: GO TO 1500
1485 PRINT INK 5;AT h,i;"G"
1490 IF h=a AND i=b THEN GO TO 4000
1498 IF i=e AND d=h THEN GO TO 3000
1501 IF j=1 THEN GO TO 1550
1510 IF b<20 THEN GO TO 1600
1520 LET j=1
1530 LET k=a: LET l=31
1550 LET k1=k: LET l1=1
1555 LET l=l-1
1560 IF a<k THEN LET k=k-.2
1565 IF a>k THEN LET k=k+.2
1570 PRINT AT INT k1,l1;" "
1580 PRINT INK 3;AT INT k,l;"M"
1590 IF l=0 THEN LET j=0: PRINT AT INT k,0;" ": LET l=10: LET k=10
1595 IF a=INT k AND l=b THEN GO TO 4000
1599 IF l=e AND d=INT k THEN GO TO 3500
1990 GO TO 1000
2000 PRINT INK 6;AT d,e;"I"
2010 BEEP .03,0
2020 IF b$(a+1)="G" THEN LET sc=sc+10: BEEP .1,45: LET b$(a+1)=" ":
LET f=f+1: PRINT AT 8,7;sc
2030 PRINT AT d,e;" "
2040 LET c=0
2090 GO TO 1300
3000 PRINT INK 6;AT d,e;"I"
3010 BEEP .1,45
3020 PRINT AT d,e;" "
3030 LET c=0: LET g=0
3040 LET sc=sc+INT (RND*3+3)*10
3050 PRINT AT 8,7;sc

```

```

3060 LET h=0: LET i=0
3080 LET d=2: LET e=2
3090 GO TO 1500
3500 PRINT AT d,e;"I"
3510 BEEP .1,45
3520 PRINT AT d,e;" "
3530 LET j=0: LET c=0
3540 LET sc=sc+100: PRINT AT 8,7;sc
3550 LET k=10: LET l=10: GO TO 3080
4000 PRINT INK li;AT li,0;"LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL"
4002 PRINT AT li,li*2;"MISSIONE CONCLUSA"
4005 LET li=li+1
4010 LET fu=32
4040 FOR z=60 TO 0 STEP -5: BEEP .1,z: NEXT z
4080 IF li=7 THEN GO TO 5000
4090 GO TO 900
5000 PRINT AT 14,6; FLASH 1; INK 2;"MISSIONE FALLITA"
5005 LET a$="          PER FARE UN'ALTRA PARTITA PREMI UN TASTO QUAL
      SIASI....."
5010 BEEP .05,20: BEEP .05,30
5012 LET a$=a$(2 TO LEN a$)+a$(1)
5017 PRINT AT 21,0; INK RND*6+2;a$(1 TO 32)
5020 IF INKEY$="" THEN GO TO 5010
5030 RUN
6000 PRINT AT a,30;" "
6005 IF fu>23 THEN PRINT AT li,0; INK li;"LLLLLLLLLLLLLLLLLLLLLLLLLLLL
      LLLLLL": LET fu=32: GO TO 6050
6010 FOR z=fu TO fu+8
6020 LET fu=fu+1: PRINT AT li,z; INK li;"L"
6030 NEXT z
6050 LET sc=sc+50: PRINT AT 8,7;sc
6080 LET b=2
6090 RETURN
9000 LET a$=a$(2 TO 32)+a$(1)
9010 LET b$=b$(2 TO 32)+b$(1)
9020 LET c$=c$(2 TO 32)+c$(1)
9090 RETURN
9100 PRINT INK 5;AT 18,0;b$
9110 PRINT a$
9120 PRINT AT 10,0;c$
9190 RETURN
9500 LET z$="{2SG4}{2SG5}{SG4}{G5}{G2}{SG5}{SG4}{SG5}{SG1}{2G5}{G2}
      {SG5}{2SG4}{SG5}{SG4}{G7}{G5}{G7}{G5}{G2}{SG4}{2G5}{G6}{3G5}
      {G3}": LET y$="{4SG5}{G5}{G8}{5SG5}{G7}{G5}{G2}{4SG5}{SG4}{G3}
      {2G5}{G8}{SG5}{3G5}{G7}{3G5}{G2}": LET x$="{4G2}{G3}{G1}{2G2}
      {G3}{2G2}{2G1}{6G2}{G8}{G1}{G3}{G1}{G2}{G3}{2G1}{G3}{2G1}{2G3}
      {3G2}PREMI UN TASTO{2G2}{G8}{G1}{G3}{G1}{G2}{G3}{2G1}{G3}{2G1}
      {2G3}"
9501 PRINT AT 21,3;"PREMI UN TASTO PER GIOCARÉ"
9502 PRINT AT 10,0;c$(1 TO 32)
9503 PRINT AT 18,0; INK 5;b$(1 TO 32)
9504 PRINT a$(1 TO 32)
9510 LET z$=z$(2 TO LEN z$)+z$(1)
9515 BEEP .004,INT (RND*30+20)
9520 LET y$=y$(2 TO LEN y$)+y$(1)
9530 LET x$=x$(2 TO LEN x$)+x$(1)
9540 PRINT INK 4;AT 2,0;z$(1 TO 32).
9550 PRINT INK 5;y$(1 TO 32)
9560 PRINT INK 6;x$(1 TO 32)
9570 IF INKEY$<>"" THEN GO TO 9600
9590 GO TO 9510

```



# Tombola

16/48K

Ci stiamo sempre più allontanando dalle antiche tradizioni. Proviamo a fare un passo indietro, giochiamo a "TOMBOLA"! Non serve nessuno per estrarre i numeri, ci pensa il nostro fidato Spectrum (lui non bala!).

L'unica cosa che occorre sono le cartelle, che dovrebbero esserci, magari in solaio, ma senz'altro ci sono.

Passiamo ora alla descrizione del programma, che senz'altro salveremo con - SAVE "TOMBOLATA" LINE 1.

In pratica il programma presenta 2 sezioni distinte: la prima è la fase della ricerca del numero e della sua

visualizzazione nel tabellone luminoso; la seconda realizza invece l'ingrandimento del numero estratto.

Problemi non dovrebbero essercene, comunque raccomando di fare attenzione poiché un solo errore potrebbe anche accendere il numero sbagliato sul tabellone, con conseguente caos tra i giocatori che non sapranno più quali sono i numeri esattamente usciti.

Non ho spiegato come si gioca a tombola perché lo ritengo superfluo, visto che gioco più tradizionale di questo non c'è.

Buon divertimento.

```
1 REM      *TOMBOLATA* 1984                      by Franzini luciano
2 PRINT AT 10,5; FLASH 1;"FERMA IL REGISTRATORE"
5 PAUSE 200
6 PAPER 6: CLS : BORDER 6: PAPER 4: LET a1=0
7 PRINT AT 2,0; PAPER 7; INK 1; BRIGHT 1;"      TOMBOLATA
  "
9 LET A2=1
10 DIM n(90)
26 PRINT
30 FOR t=0 TO 80 STEP 10
40 FOR i=1 TO 10
45 LET A2=A2+0.5
50 PRINT TAB ((i*3)-2);i+t;
60 NEXT i
62 PRINT " ";
65 BEEP .1,A2
80 PRINT
90 NEXT t
95 PRINT #0; INK 2; BRIGHT 1;"PREMI ENTER PER ESTRARRE UN NUM.PREMI
  S PER UNA NUOVA TOMBOLATA."
```

```

96 DRAW 0,175: DRAW 255,0: DRAW 0,-174: DRAW -255,0
99 PAUSE 0
100 RANDOMIZE
110 LET q=INT (RND*90)+1
120 IF n(q)=1 THEN GO TO 110
140 GO SUB 4000: GO SUB 1000
150 LET x=INT (q/10)
160 IF q/10=INT (q/10) THEN LET x=x-1
170 LET y=q-(x*10)
200 PRINT AT x+4,3*y-2; PAPER 6; BRIGHT 1;q
201 IF a1=89 THEN PRINT AT 20,0;"SONO GIA' USCITI TUTTI I NUMERI!":
    PRINT #0;AT 1,0;"
    ": FOR t=1 TO 1000: NEXT t: RUN 6
204 LET a1=a1+1
210 GO TO 110
1000 LET n(q)=1
1050 PAUSE 0
1064 IF INKEY$="s" OR INKEY$="S" THEN RUN 6
1065 IF INKEY$<>CHR$ 13 THEN GO TO 1050
1075 BEEP .1,18
1077 RETURN
4000 FOR t=1 TO 6
4001 PRINT AT 13+t,1; PAPER 6;"
4002 NEXT t
4004 LET d=0
4005 LET c=PEEK 23606+256*PEEK 23607
4010 LET x$=STR$ q
4020 IF LEN x$=1 THEN LET x$=" "+x$
4025 FOR t=0 TO 1
4030 LET pl=c+CODE x$(t+1)*8
4040 FOR g=0 TO 7
4050 LET p=PEEK (pl+g)
4060 FOR i=0 TO 7
4070 IF p/2<>INT (p/2) THEN PRINT AT 13+g,14-i+(t*8); INK 2;
    BRIGHT 1;"{SG8}"
4080 LET p=INT (p/2)
4085 NEXT i
4090 NEXT g: NEXT t
4091 BEEP .5,23
4093 BEEP .6,19
4110 POKE 23692,-1
4120 RETURN

```



# Guida per l'input dei programmi sullo ZX Spectrum

Come è noto, lo Spectrum è provvisto di 2 serie di tasti grafici: una prima serie di 16 caratteri grafici predefiniti (i tasti numerici da 1 a 8 e gli stessi "shiftati") e una serie di caratteri definibili dall'utente nell'ambito di un programma (le lettere da A a U). In entrambi i casi, per ottenere i caratteri desiderati occorre entrare in modo grafico (cursore contrassegnato dalla lettera G lampeggiante) premendo contemporaneamente i tasti CAPS SHIFT e 9.

Nei nostri listati i caratteri grafici predefiniti sono indicati da una G e dal numero corrispondente al tasto che occorre digitare, il tutto racchiuso tra due parentesi graffe.

A esempio {G4} significa che occorre digitare il tasto 4, con il cursore in modo grafico.

Analogamente la codifica SG, seguita da un numero da 1 a 8, significa che occorre digitare il relativo tasto numerico premendo contemporaneamente il tasto CAPS SHIFT.

Ad esempio quando si trova la codifi-

ca {SG2}, occorre premere il tasto 2 contemporaneamente al tasto CAPS SHIFT, ovviamente con il cursore in modo grafico. In entrambi i casi precedenti, quando un simbolo grafico deve essere digitato più volte, i caratteri G o SG della codifica sono preceduti da un numero che specifica quante volte va premuto il tasto grafico indicato.

Così ad esempio {8G5} significa che il tasto grafico 5 va digitato 8 volte; analogamente {4SG1} significa che il tasto grafico 1, premuto insieme a CAPS SHIFT, deve essere battuto 4 volte. I caratteri grafici definibili (le lettere da A a U in modo grafico) hanno una codifica semplificata: la lettera corrispondente, sottolineata.

Quando in un listato viene presentata, ad esempio, una A sottolineata, occorre entrare in modo grafico (al solito premendo contemporaneamente i tasti CAPS SHIFT e 9) e quindi digitare semplicemente il tasto che contrassegna la lettera A.

Quando leggete	Premete	Vedrete
{G1}		
{G2}		
{G3}		
{G4}		
{G5}		
{G6}		
{G7}		
{G8}		
{SG1}		
{SG2}		
{SG3}		
{SG4}		
{SG5}		
{SG6}		
{SG7}		
{SG8}		

Se non siete già in modo G, premete contemporaneamente CAPS SHIFT e 9

Se dovete usare il modo G, premete contemporaneamente CAPS SHIFT e 9

Quando leggete	Premete	Vedrete
A		
B		
C		
D		
E		
F		
G		
H		
I		
J		
K		
L		
M		
N		
O		
P		
Q		
R		
S		
T		
U		

Simbolo grafico definito nel programma in uso.

Se non siete già in modo G, premete contemporaneamente CAPS SHIFT e 9

Se dovete usare il modo G, premete contemporaneamente CAPS SHIFT e 9



GRUPPO EDITORIALE

**JACKSON**

DIVISIONE LIBRI

# Le strenne firmate Jackson

**COMPUTERGRAPHIA**

Joan Scott

In 176 meravigliose e coloratissime pagine, uno splendido viaggio attraverso le immagini elaborate con le tecniche più avanzate di computer-graphics.

Cod. 802P Lire 40.000

**COFANETTI SPEEDY COMPUTER**

L'informatica attraverso i testi e le immagini studiate per i bambini può diventare facile anche per ... mamma e papà.

Cod. CU001 Lire 45.000

Cod. CU002 Lire 45.000

**COMPUTER IMAGE**

Mauro Sabatini

Chi è ancora convinto che il computer limiti le possibilità creative dell'uomo, può scorrere nel libro, le splendide immagini di computer-art a colori e ... convincersi del contrario.

Cod. CI231 Lire 40.000

**ROMANZO ROSA CON IL C64**

Marco Buxton, Francesco Davini, Stefania D'Ambrò

Sei un tipo romantico? Ti piacciono i romanzi?

Il computer, attraverso un simpatico programma scritto per Commodore 64, collabora con te e con ... nella costruzione di un Vostro "Romanzo rosa".

Con floppy disk.

Cod. CC130 Lire 40.000

**IMPARA IL BRIDGE CON IL COMPUTER**

Claudio Trinchillo

Imparare il bridge con l'aiuto del computer diventa un piacevole passatempo. Un programma interattivo elaborato su IBM e su C 64 ti conduce passo passo a diventare ... quasi un esperto di bridge. Poi con un po' di applicazione ...

Con floppy disk per C64

Cod. CC219 Lire 50.000

Con floppy disk per IBM

Cod. CC239 Lire 50.000

**Puoi trovare le strenne Jackson  
nelle migliori librerie**